ADMINISTRATION GUIDE | PUBLIC
SAP Adaptive Server Enterprise 16.0 SP03
Document Version: 1.0 – 2020-03-04

Workload Analyzer Users Guide

THE BEST RUN SAP

Content

1 Workload Analyzer

Workload analyzer option for the SAP Adaptive Server Enterprise allows you to capture, analyze, and replay a production workload non-disruptively. You can then use the captured workload to diagnose problems, and understand and manage configuration changes.

Workload analyzer also allows you to replay captured workloads to measure and analyze application performance under different conditions.

Use the workload analyzer option to:

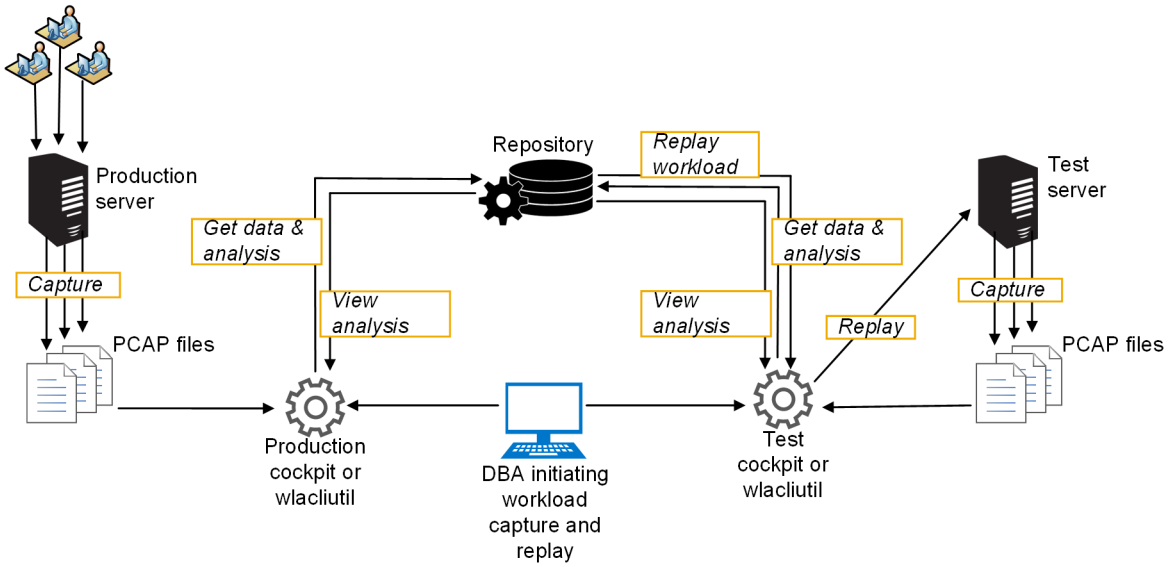
- Identify problematic queries, such as queries with a long response time.
- Identify client activity patterns, such as the number of requests per IP address.
- Measure the performance of captured workloads in different server configurations.
- Compare query and overall workload performance between different server configurations.
- Evaluate database upgrades and understand the benefits from new options.
- Diagnose product problems by replaying functionality in a controlled environment.
- Pinpoint potential issues such as why certain queries are running slowly.
- Determine the longest running query.
- Test new features and run them against a captured workload to verify performance.
- Capture the workload on the target server that is running the replay to compare performance against its original replay.

You can use the workload analyzer option either with SAP ASE Cockpit in GUI mode or with `wlacliutil`, a command line utility. This guide focuses on using the `wlacliutil` utility. For information on using the workload analyzer in the cockpit environment, see *SAP Adaptive Server Enterprise Cockpit > Monitor SAP ASE > Monitor Captured Workloads*. Regardless of the method you choose to use the workload analyzer, follow the steps in this guide to first set up the repository server and install the repository database.

2 Overview

Describes the general capture-analysis-replay workflow.

The following diagram illustrates a general workflow for workload capture, analysis and replay:



1. Initiate capture of the production workload. You can use the SAP ASE cockpit, the `wlacliutil` command line utility, or the `dbcc workload_capture` command.
2. When capture starts, the production server writes the raw workload data into one or more packet capture (PCAP) files. When the capture is finished, the server generates the PCAP files with the captured workload.
3. The workload analyzer loads the server-generated PCAP files into the repository database and generates analytical and statistical information about the captured workload.
4. Analytical and statistical information becomes available for view in the SAP ASE cockpit or `wlacliutil` utility.
5. After a workload capture is generated and the file is available from the repository server, you request a workload replay from the SAP ASE cockpit or `wlacliutil` utility.
6. Workload analyzer generates a replay of a captured workload.
7. The captured replay becomes available for view and analysis from the SAP ASE cockpit or `wlacliutil` utility.

For more information, see https://www.youtube.com/watch?v=w02qo9WHLsw .

> **Note**
>
> The SAP ASE cockpit provides some analytical and statistical information, and comparison results that are not available in the `wlacliutil` utility.

3 Requirements

To use the workload analyzer feature, make sure you meet these requirements.

Requirements	Description
ASE_WORKLOADANALYZER license	Purchase the ASE_WORKLOADANALYZER license so you can install the workload analyzer option on your SAP ASE production server, from where you plan to capture the workload. With the license, you can enable the feature by setting the `enable workload analyzer` configuration parameter to `1` (on). This configuration is dependent on the existence of the ASE_WORKLOADANALYZER license.
`enable workload analyzer` configuration parameter	After obtaining the ASE_WORKLOADANALYZER license, enable the feature by setting the `enable workload analyzer` configuration parameter to `1` (on)
Repository server	In addition to the SAP ASE server with the workload you plan to analyze, install an additional server from which to perform your workload replay and analysis. Having a separate nonproduction server helps to avoid performance contention issues. The license for the Workload Analyzer feature includes entitlement to use an additional SAP ASE server for the purpose of hosting the Workload Analyzer repository database. The SAP ASE server license provided with the Workload Analyzer feature includes all the capabilities required to support the Workload Analyzer repository database.
Semantic partitioning	The `sybcatdb` database that is installed on the repository server requires semantics-based partitioning, a licensed feature.
SAP ASE cockpit or `wlacliutil` utility	Use either the SAP ASE cockpit or `wlacliutil` utility to capture, analyze, and replay workloads. If you want to run a workload replay and analysis from an SAP ASE cockpit environment, enable SAP ASE cockpit when you install your SAP ASE server. See the *SAP Adaptive Server Enterprise Cockpit* documentation for details on how to use cockpit.
Synchronizing SUIDs	As part of the overall server and database environment, the replay server and the copy of the application database on that server — that is, the database used by the operations contained in the captured workload that will be replayed, and not system databases such as `master` or `model`, or from other databases that might reside on your SAP ASE server that you have loaded the application database onto — should have the same characteristics as on the source production server. The application login must be a user in the application database, and the SUID for that user must be the same in the master and application databases. This requirement is not unique to Replay. Synchronizing SUIDs is a common administrative requirement that our customers often encounter any time a databases is restored to a new server.
Operating system logins	Whenever you use this option, make sure to start both SAP ASE and cockpit using the same operating system login. Cockpit must be able to read the pcap files that SAP ASE generates, and cockpit creates the output directories for the pcap files and ASE must be able to write to those directories. Because access to the directories and files are automatically restricted to the operating system login that creates them, SAP ASE cockpit

Requirements	Description
	cannot perform a workload analysis unless cockpit is running under the same operating system login as the SAP ASE server.
Setting login default database	Make sure to configure the attributes of all logins that use the application to explicitly be able to use the application database.

4 Creating the Repository Server

In order to avoid performance contention issues when you run an analysis of a captured workload, install a second SAP ASE that is a 16KB page server.

The repository server is a separate, nonproduction server where your repository database will reside, and from where you perform your workload replay and analysis.

Use the instructions in the SAP ASE installation guide for your platform, with the following that are specific to the repository server:

- Specify the server as a 16K page size. If you use any other page size for the repository server, the `installsybcatdb` script fails.
- Configure the repository server to `utf8`, even if the workload does not use `utf8`. Without this setting, the `installsybcatdb` installation script fails.

Related Information

Installing the Workload Analyzer Repository Database [page 9]

4.1 The sybcatdb Database on the Repository Server

Before you can perform a workload capture, create the database devices for the `sybcatdb` data and log segments and create the `sybcatdb` database on those devices.

- Make sure the repository server has been created with a 16KB page size, with UTF8 as your default character set.
- Use `sp_dboption` to set `select into/bulkcopy/pllsort` option to `true`.
- Use `sp_configure "lock scheme"` to set the locking scheme as `datarows`.

Create the data and log segments in separate devices. The size of the data segment is dependent on the workloads saved in the repository database.

See Determining the Size for sybcatdb [page 8] for advice on setting the size of the `sybcatdb` database.

Related Information

Running the installsybcatdb Script [page 9]

4.1.1 Determining the Size for sybcatdb

Although the sybcatdb database only requires 50MB to successfully run the installsybcatdb script, this size is too small to contain any reports for workload analyses. Consider the following to calculate the optimal size in which to extend the size of sybcatdb.

In most cases, you can set the repository database data segment as four times the total size of all PCAP files that will be loaded into the repository.

Although the space required in the `sybcatdb` database depends on the number of requests and other characteristics of the captured workload, an initial size of 500 to 1000 MB may be sufficient to store an initial captured workload and to gain experience with the requirements for capturing your workload. The space required to store multiple larger captured workloads may be much greater than this.

In scenarios where there are many short queries or many dynamic SQL statements, increase the data segment size to at least six times the total PCAP file size.

Keep the log segment size to at least 2 GB to run workload analyses and replays.

To prevent log-full issues during analysis or replay of a very large workload, increase the log size of `sybcatdb` and `tempdb`.

> **i** Note
>
> Any threshold procedure in `tempdb` you create will disappear when you restart the server and `tempdb` is re-created. For this reason, always re-create the threshold procedure after a server restart.

5 Installing the Workload Analyzer Repository Database

Run the installation script to create the workload analyzer repository database.

The `sybcatdb` database holds the captured workload and the results of analysis that are displayed in the workload dashboard in SAP ASE cockpit. The captured workload, saved initially as a PCAP file, contains the original raw TDS packets and other metadata that is used during replay to send the application requests to the target ASE server during replay and to order those requests correctly.

In order for the data in the PCAP files to be processed, run the `installsybcatdb` script on your repository server. The script then creates the tables, views, indexes, and stored procedures in the sybcatdb database.

5.1 Running the installsybcatdb Script

The installsybcatdb script creates the tables, views, indexes, and stored procedures for the sybcatdb database on your repository server.

Prerequisites

- The SAP ASE server on the repository server uses 16K page size.
- The UTF-8 character set is installed as the default character set on the SAP ASE you installed on the repository server.
- The locking scheme is set as `datarows`.
- The `enable semantic partitioning` configuration parameter is set to `1`.
- The `sybcatdb` database already exists on the SAP ASE on the repository server.
- The user who executes the script has the `sa_role` role.

Procedure

1. Start SAP ASE on your repository server.
2. Go to the `scripts` directory at the following:

Option	Description
UNIX	`$SYBASE/COCKPIT-4/plugins/ASEMAP/scripts`
Windows	`%SYBASE%\COCKPIT-4\plugins\ASEMAP\scripts`

3. Use `isql` to log in to the SAP ASE server and run the script, where `<password>` is the password string, and `<server_name>` is the destination server for the database:

Option	Description
UNIX	`isql -Usa -P<password> -S<server_name>` `-i$SYBASE/COCKPIT-4/plugins/ASEMAP/scripts/installsybcatdb`
Windows	`isql -Usa -P<password> -S<server_name>` `-i %SYBASE%\COCKPIT-4\plugins\ASEMAP\scripts\installsybcatdb`

6 Configuring the Repository Server

Set the server configuration settings to run a captured workload analysis.

Unless otherwise specified, use `sp_configure` to set these configuration settings.

Configuration Settings	Minimum Values
`default data cache`	Use `sp_cacheconfig` to set this value to 500MB. The default value is 8MB. Although it is possible for small workloads to run with the default value of 8MB, this value may cause larger workloads to run very slowly.
`default language id`	Set to NULL, to specify English as the default language for the repository database. The default value is `0`
`disable varbinary truncation`	Set to `1`. The default value is `0`. The workload analyzer may not work if you do not change the value of `disable varbinary truncation`, because values could have their trailing zeros truncated (for example 0x0f01007d000000001000000032136a636f6e6e6563745f696d706c696369745f310000000 could be truncated to 0x0f01007d000000001000000032136a636f6e6e6563745f696d706c696369745f31). This could cause an unexpected error in the TDS parser.
`heap memory per user`	Set to `49152`. The default value is `4096`.
`lock scheme`	Set as `datarows`; the default is `allpages`. Keeping the lock scheme default of `allpages` could cause a deadlock issue during analysis, since all temporary tables that are created during analysis will use the allpages lock scheme.
`max memory`	Set to `1150976`. The default value is platform-dependent.
`max online engines`	Set to `4`. The default value is `1`. **i Note** If you are using threaded mode, in addition to setting `max online engines`, change the number of threads in the default pool to "4" by using the following command: `alter thread pool syb_default_pool with thread count = 4` See *Reference Manual: Commands > alter thread pool* for more information about altering a thread pool.
`max parallel degree`	Keep the default value of `1`. This configuration parameter specifies the server-wide maximum number of worker processes allowed per query.
`max utility parallel degree`	Set to `8`. The default value is `1`.

Configuration Settings	Minimum Values
	i Note This configuration parameter differs from `max parallel degree`. The `max utility parallel degree` parameter specifies the server-wide maximum number of worker processes allowed per query used by the `create index with consumers` and `update stats with consumers` commands.
`number of locks`	Set to `500000`. The default value is `10000`. If out-of-lock errors occur during analysis or replay, then increase the configured number of locks.
`number of user connections`	Make sure that there are at least 21 connections available for analyzing a workload. The default is 25
`number of worker processes`	Set to `8`. The default value is `0`.
`procedure cache size`	Set to `100000`. The default value is `14000`.

Set the size of the database (`tempdb`) to 500MB. The default is 124MB.

See *Reference Manual: Configuration Parameters* for complete details on these parameters.

6.1 Add a Workload User Login

Create a workload user login after you configure the repository server.

Context

After configuration is complete, create an SAP ASE server login that the cockpit uses to connect to the workload repository server, and make that login a user in the `sybcatdb` database. After the user is created, cockpit can connect to the repository database.

Procedure

1. Use the `create login` command to create a new user, such as "`workload_user`":

   ```
   create login workload_user with password itsASecret default database sybcatdb
   ```

2. Use the `sp_addalias` system stored procedure to allow the `workload_user` login to be known in the `sybcatdb` database as the database owner:

   ```
   1> use sybcatdb
   2> go
   ```

```
1> sp_addalias 'workload_user', dbo
```

If you do not perform this step, the login will require an "sa_role" to use the `sybcatdb` database regardless of whether granular permission is enabled or not.

6.2 Temporary Database Configuration

Set the `abort tran on log full` database option to `false` for the Workload Analyzer Repository server temporary database used by the Workload Analyzer login.

In some cases, the transaction log space in the temporary database can be temporarily exhausted. If a long-running workload analysis is in progress when this occurs, processing is temporarily suspended until space becomes available in the temporary database transaction log. In this case, if the `abort tran on log full` option is enabled, analysis is terminated rather than temporarily suspended and analysis would need to restart.

> **i** Note
>
> If the log full condition is not temporary, analysis is suspended until additional log space is added to the temporary database. Check the Workload Repository Database server error log for messages indicating that transactions have been suspended. And if needed, use the `alter database` command to add space to the temporary database.

6.3 sybcatdb Database Configuration

Set the `abort tran on log full` database option to `false` for the `sybcatdb` database in the Workload Analyzer repository server.

In this case, if a long-running workload analysis is in progress when the transaction log space in the `sybcatdb` database runs out of space, the workload analysis is suspended until additional log space is added to the `sybcatdb` database. If the `abort tran on log full` option is enabled (set as `true`), the workload analysis is terminated rather than suspended and needs a restart later.

Check the Workload Repository Database server error log for messages indicating that transactions have been suspended. Use the `alter database` command to add additional log space.

7 Tuning Tips

Improve analysis performance by optimizing the repository database.

While you can perform a successful analysis if you correctly configure the `sybcatdb` database, these additional settings may increase analysis speed.

- Consider binding the `sybcatdb` and `tempdb` databases to separate data caches.
- Increase the engine number to 16 or more.
- Set the following configuration parameters using `sp_configure`:

Parameter	Setting
`number of oam trips`	`20480`
`number of index trips`	`20480`
`lock hashtable size`	`524288`
`default network packet size`	`8192`
`max network packet size`	`65024`
`number of sort buffers`	`3000`
`procedure cache size`	`192000`
`allocate max shared memory`	`1`
`max memory`	Configure this based on the above values.

See *Reference Manual: Configuration Parameters* for complete details on these parameters.

8 Workload Analyzer Command Line Utility

The Workload Analyzer command line utility – `wlacliutil` – is another tool that you can use to capture, analyze, and replay workloads.

Use `wlacliutil` either interactively or with a response file to perform Workload Analyzer tasks.

8.1 wlacliutil in Interactive Mode

In interactive mode, `wlacliutil` provides commands and menu prompts to help you make selections and enter proper values for an option.

The utility rejects invalid entries and displays warnings or error messages when you make improper selections or enter an invalid value.

8.1.1 Starting wlacliutil

Use `wlacliutil` to connect to a repository server in interactive mode.

To start `wlacliutil`, run:

```
wlacliutil -S<repository_server_name> -U<repository_username> -
P<repository_password> -I<interfaces_file>
```

- The `-S` and `-U` options are mandatory.
- The `-P` option is optional. If you don't include `-P`, `wlacliutil` prompts you to enter a password for the repository database once the utility has started.
- The `-I` option is optional. If you do not specify `-I`, `wlacliutil` looks for a file named interfaces in the directory specified by your SYBASE environment variable.

Once your login is accepted, `wlacliutil` displays a prompt:

```
<repository_server_name>>
```

If you specified the server name using the `<repository_server_host_ip>:<port_number>` format, then the prompt is:

```
<repository_server_host_ip>:<port_number>>
```

This prompt indicates that `wlacliutil` has connected to this repository server. If the connection fails, `wlacliutil` displays an error message indicating the failure reason. Fix the issue and reconnect.

Run `disconnect` to disconnect from the repository server.

8.1.2 The Workload Capture

You can capture, import, view, and delete a workload capture with `wlacliutil` in interactive mode.

8.1.2.1 Capturing an SAP ASE Workload

SAP ASE Workload Analyzer captures the complete production workload without disrupting currently running SAP ASE transactions.

Prerequisites

`wlacliutil` is running in interactive mode.

Procedure

1. Enter the following command to start a workload capture and specify a name for it:

   ```
   capture start <capture_name>
   ```

   If you don't specify the capture name, then you are prompted to enter one.
2. In the wizard, enter the required information for the capture.
   a. (Optional) Add comments for this capture.
   b. Specify the capture server information:

Capture Server Information	Description
Enter the host name or IP address of ASE server:	Host name or IP address of the SAP ASE server you want to capture workload on.
Enter the port number of ASE server:	Port number of the SAP ASE server.
Enter a username:	A login on the SAP ASE server. This login requires the following permission: ○ When `enable granular permission` is disabled, the login must have the SA or SSO role. ○ When `enable granular permission` is enabled, this login must have the `set tracing any process`.

Capture Server Information	Description
Enter a password:	The password for the login.
Do you want to use SSL?	Specifies whether to enable SSL for the connection to the SAP ASE server on which the capture is running.
Enter SSL trust store file:	Location of the SSL trust store file. The trust store file includes the public key used to establish SSL-enabled connections. Import the public key into this trust store file with the following command: `$SAP_JRE/bin/keytool.exe -import -keystore <TrustStore> -storepass changeit -file trusted.txt`

c. Provide the following capture options information:

Capture Options	Description
Apply a login filter:	Specifies which login(s) to include in or exclude from the capture. The default value is `N`, which includes all logins in the capture. If you specify `Y`, then a list of all available logins on the SAP ASE server from which the workload is captured appears. Enter the required login name(s) to include or exclude.
Apply an application filter:	Specifies which application(s) to include in or exclude from the capture. The default value is `N`, which includes all applications in the capture. If you specify `Y`, then enter the required application name(s) to include or exclude.
Enter capture duration (Seconds):	Specifies the capture time limit in seconds. The capture stops when the capture time exceeds the specified limit. The default value is `0`, which means there is no time limit. You can end a capture at any time by using the `capture stop` command.
Enter workload storage location:	Specifies where to store captured workload files to.
Save all TDS response data:	Specifies whether to capture all response TDS data (rather than the last response packet for each request).

Capture Options	Description
Enter file size limit (MB):	Specifies whether to stop the capture when the file reaches a specific size.
Enter overhead limit (%):	Specifies the acceptable overhead percentage.

Once you provide all required information, the wizard displays a capture summary for you to confirm the information:

```
--------------------Capture [xxxxx] Summary--------------------
Capture Name              : xxxx
Capture Comments          : xxxx
Server Name               : <host>:<port>
Use SSL connections       : Yes
SSL TrustStore File       : xxxx
Workload Storage Location : xxxx
Include Logins            : INCLUDE: ALL_LOGINS
Include Applications      : INCLUDE: ALL_APPS
Stop On PCAP File Size    : Unlimited
Stop on Overhead Limit    : Unlimited
Save ALL TDS Response     : NO
Start([s]), Edit(e) or Cancel(c)?
```

d. Enter one of the following options to proceed:
   - `s` (default) or `Enter` – starts the workload capture.
   - `e` – modifies the provided information.
   - `c` – cancels the capture.

Once the capture starts, `wlacliutil` periodically displays the capture status. For example:

```
------------------Capture [xxxx] Status------------------
Server      : <host>:<port>
StartTime   : 2019-01-22 15:39:59
Elapsed Time :[00:00:01].  Overhead Percentage:[0%].  Captured PCAP File Size:
[24 Bytes  ].  Captured Packages:[0].
```

3. (Optional) Run `capture status` to check the capture status.
4. If you set the capture duration to 0, run `capture stop` to stop the workload capture.

Results

When the capture process is complete, `wlacliutil` displays a report on this capture. For example:

```
Capture [xxxx] has been stopped successfully.
Capture Name:                 xxxx
Capture Comments:             xxxx
Capture Server:               <host>:<port>
Capture State:                Stopped
Capture Start Time:           2019-01-22 15:39:59
Capture End Time:             2019-01-22 15:48:25
Capture Duration:             00:08:26
Capture Stop Reason:          Request
Capture Packets:              572
Capture PCAP File Size:       251.01 KB
Capture Overhead Percentage: 0%
Capture Files:
```

```
/opt/sap/ase_20181123_145549/ase.0.pcap
/opt/sap/ase_20181123_145549/ase.1.pcap
```

You can now analyze the workload.

8.1.2.2 Importing a Captured Workload

Import previously created captured workloads into a repository database.

Prerequisites

`wlacliutil` is running in interactive mode.

Procedure

1. Enter the following command to import the required workload:

   ```
   capture import <capture_name>
   ```

   If you don't specify the capture name, you are prompted to enter one.
2. In the wizard, enter the required information for the capture.
   a. (Optional) Add comments for this capture.
   b. Enter the location of the PCAP files.
   c. If the location is accessible, `wlacliutil` displays all available PCAP files under this directory and asks you whether to import all these PCAP files. Enter one of the following options to proceed:
      - Y – imports all PCAP files under this location.
      - N – imports selected PCAP files. Enter the file names separated by a comma.
   d. Specify whether to analyze this workload automatically.
   e. `wlacliutil` displays a summary for you to confirm. Enter one of the following:
      - `i` or `Enter` – imports the capture
      - `e` – modifies the specified values
      - `c` – cancels the import

Next Steps

You can now analyze the workload.

8.1.2.3 Deleting a Captured Workload

Remove captured workloads that are no longer needed from the repository database.

Procedure

1. To delete a captured workload, run:

   ```
   capture delete <capture_name>
   ```

   If you don't specify the capture name, then you are prompted for a name.

   `wlacliutil` displays information similar to this:

   ```
   Capture Files:
          /tmp/XIYL50833394A_20190128_112634.pcap
   Will PCAP files for this capture be deleted?   [ N ]
   The following replays will be deleted automatically if you delete this capture:
           replay_demo
   Do you really want to delete this capture?  [ Y ]
   ```

2. Choose whether to delete PCAP files for this capture by typing:
   - `Y` – deletes PCAP files and all replays for this capture.
   - `N` – keeps PCAP files. You can manually delete them later.
3. At the confirmation prompt, type `Y` to delete the capture. All replays of this capture are also deleted.

8.1.2.4 Viewing a Captured Workload

View detailed information of a captured workload, including analytical information, if the workload is analyzed.

Procedure

To view the captured workload, run:

```
capture show <capture name>
```

`wlacliutil` displays information similar to this:

```
Name                    : cmdline_capture22
Status                  : Analyzed
Start Time              : 2019-01-18 17:47:15
Duration                : 00:01:23
Comments                : <Empty>
```

If the capture is analyzed, `wlacliutil` also displays analytical information like this:

```
Server Name             : 13.153.0.215:13655
Average Execution Time  : 00:00:00.011753
Number of Sessions      : 6
```

```
Number of Requests      : 230
Number of Errors        : 0
```

Running the `capture show` command without the capture name displays information of all captures. For example:

```
Name                           Status        Start Time           Duration
-----------------------------  ------------  -------------------  --------
SAPIT2_1                       Initialized   n/a                  n/a
Kerberos_21                    Analyzed      n/a                  n/a
tpcc                           Stopped       n/a                  n/a
test001                        Initialized   n/a                  n/a
tpcc1                          Initialized   n/a                  n/a
tpcc2                          Initialized   n/a                  n/a
tpcc3                          Initialized   n/a                  n/a
tpcc4                          Initialized   n/a                  n/a
tpcc5                          Initialized   n/a                  n/a
tpcc6                          Initialized   n/a                  n/a
KernelModeTest_2               Analyzed      2017-02-14 14:06:01  00:00:41
ILT_20180105_092509            Loaded        2018-01-05 16:25:09  00:04:48
ILT_20180105_095212            Loaded        2018-01-05 16:52:12  00:01:40
…
```

8.1.3 Analyzing Captured Workloads

Analyze a captured workload to get a basic capture summary, such as capture duration, number of sessions, number of requests, and number of errors.

Prerequisites

Make sure the PCAP files are accessible. If they are on a remote server, copy them to a local directory that is accessible. Otherwise, `wlacliutil` prompts you to transfer the PCAP files before you can start analyzing a capture.

Context

This version of `wlacliutil` doesn't provide advanced analytical information, such as the longest running requests and the most frequent running requests, or requests from IP, login, or application that sent the most number of requests. Use SAP ASE Cockpit to view the advanced information.

Procedure

1. To begin the analysis, run:

   ```
   analyze start <capture_name>
   ```

   If you don't specify the capture name, you are prompted to enter one.

The workload analysis begins.

2. `wlacliutil` periodically reports the analysis status. You can also run `analyze status` at any time to check the status. The status report is similar to this:

   ```
   Workload [cmdline_capture25] is in [Identifying requests]. Analysis Percentage:
   [79.17%]. Elapsed Time: [00:00:33]
   ```

3. Run `analyze stop` to stop the workload analysis if necessary.

Results

When the analysis completes, `wlacliutil` displays a report similar to the following:

```
Name                    : cmdline_capture25
Status                  : Analyzed
Start Time              : 2019-01-22 15:39:59
Duration                : 00:08:26
Average Execution Time  : 00:00:00.012370
Comments                : This is a demo.
Number of Sessions      : 6
Number of Requests      : 230
Number of Errors        : 0
Server Name             : 11.163.1.235:13655
```

8.1.4 The Captured Workload Replay

Run replays of captured workloads to measure and analyze application performance.

8.1.4.1 Creating a Replay

Create a replay of a previously created captured workload.

Prerequisites

A previously created captured workload, which has been analyzed, exists.

Procedure

1. To create a replay, run:

   ```
   replay start <replay name>
   ```

   If you don't specify the replay name, you are prompted to enter one.

2. In the wizard, enter the required information for the replay
   a. (Optional) Add comments for this replay.
   b. Specify the following replay server information:

Replay Server Information	Description
Enter the host name or IP address of ASE server:	Host name or IP address of the SAP ASE server you want to replay the workload on.
Enter the port number of ASE server:	Port number of the SAP ASE server.
Enter a username:	A login on the SAP ASE server. This login requires the following permission: ○ When `enable granular permission` is disabled, the login must have the SA or SSO role. ○ When `enable granular permission` is enabled, this login must have the `set tracing any process` permission.
Enter a password:	The password for the login.
Do you want to use SSL?	Whether to enable SSL for the connection to the SAP ASE server you want to replay the workload on. If you did not capture a workload using SSL, then do not use SSL during replay. The average execution time for queries is longer when using SSL, which could impact the performance of a replay.
Enter SSL trust store file:	Location of the SSL trust store file. The trust store file includes the public key used to establish SSL-enabled connections. Import the public key into this trust store file with the following command: `$SAP_JRE/bin/keytool.exe -import -keystore <TrustStore> -storepass changeit -file trusted.txt`

   c. Specify the following replay options:

Replay Options	Description
Enter a capture name:	Name of the capture you want to replay.

Replay Options	Description
Apply a login filter:	Specifies which login(s) to include in or exclude from the capture. The default value is `N`, which includes all logins in the replay. If you specify `Y`, then a list of all available logins in the captured workload appears. Enter the required login name(s) to include or exclude.
Apply an application filter:	Specifies which application(s) to include in or exclude from the capture. The default value is `N`, which includes all applications in the replay. If you specify `Y`, then enter the required application name(s) to include or exclude.
Do all logins have the same password on replay server?	Specifies whether all logins have the same password on the repaly server. If you enter: ○ `Y`, enter the generic password. ○ `N`, enter the password for each login.
Will login failed session in source workload be included?	Specifies whether to replay failed login sessions.
Specify the replay speed for this replay:	Changes the default value of 1.0 to change the speed at which to run the replay. The values range from 0.1 to 10.0. A higher value means a higher replay speed. However, 0 means the fastest speed.
Will the workload be captured during replay?	If you keep this default, you can also specify the location in which to save the PCAP files.
Enter workload storage location:	Specifies where to store the PCAP files.
Will new connections which are not in source workload be included?	Specifies whether to capture new connections outside the original capture.
Will time of replay server be reset to source workload start time?	When date and time adjustment is not used during replay (default setting), the date and time on the replay SAP ASE server will be the actual time at which the replay occurs. If you specify `Y`, when the replay begins, the date and time of the replay SAP ASE server are set to the time at which the capture originally started.

Once you provide all required information, the wizard displays a summary of replay options for you to confirm the information:

```
--------------------Replay [xxxx] Summary--------------------
Replay Name                    : xxxx
```

```
Replay Comments                : xxxx
Source Workload                : xxxx
Replay Server                  : <host>:<port>
Include Logins                 : <all>
Include Applications           : <all>
Workload Storage Location      : xxxx
Replay Failed Sessions         : No
Replay Speed                   : 1
Capture Workload During Replay : Yes
Include New Connections        : No
Reset Server Time              : No
Use SSL Connections            : Yes
SSL Trust Store File           : c:\work\asecat\wkload_cli\cacerts
```

d. Enter one of the following options to proceed:
   - `s` or `Enter` – starts the replay.
   - `e` – modifies some replay options.
   - `c` – cancels the replay.

3. Once started, `wlacliutil` periodically displays the replay status. For example:

```
------------------Replay [xxx] Status----------------------------
Source Workload   :xxxx
Start Time        :2019-01-22 16:51:13
Total Requests    :230
Execution Duration :[00:00:10].  Replay Progress:[96.00%]  Replayed Requests:
[24].
```

   You can run `replay status` at any time to check the status if necessary.

4. (Optional) Run `replay stop` to stop the replay.

   The replay might be incomplete when you manually stop it, which can affect the replay comparison result.

Next Steps

You can now analyze the replay.

8.1.4.2 Viewing a Replay

View detailed information of a workload replay, including analytical information if the replay is analyzed.

Procedure

To view the replay, run:

```
replay show <replay name>
```

`wlacliutil` displays information similar to this:

```
Name                    : cmdline_replay7
Capture Name            : cmdline_capture22
```

```
Status                  : Analyzed
Start Time              : 2019-01-18 17:54:46
Duration                : 00:01:42
Comments                : <Empty>
```

If the replay is analyzed, `wlacliutil` also displays analytical information like this:

```
Average Execution Time : 00:00:00.013822
Number of Sessions     : 1
Number of Requests     : 182
Number of Errors       : 0
Replay Server          : 10.173.0.245:13655
```

Running the `replay show` command without the replay name displays information of all replays. For example:

```
Name                            Status        Start time           Duration
------------------------------  ------------  -------------------  --------
DYNTEST_1_REPLAY_1              Analyzed      2018-04-25 18:27:58  00:01:40
DYNTEST2_RELAY1                 Stopped       2018-04-25 18:32:03  00:02:32
DYNTEST_1_REPLAY_2              Analyzed      2018-04-25 18:48:49  00:01:39
DYNTEST_1_REPLAY_3              Analyzed      2018-04-25 18:58:20  00:01:40
DYNTEST_3_REPLAY_1              Analyzed      2018-04-25 23:23:47  00:03:31
```

...

8.1.4.3 Deleting a Replay

Remove replays that are no longer needed from the repository database.

Procedure

1. To remove a replay, run:

   ```
   replay delete <replay_name>
   ```

   If you don't specify the replay name, you are prompted to enter one.

   `wlacliutil` displays information similar to this:

   ```
   Capture Files:
          / tmp/XIYL50833394A_20190128_122202.pcap
   Will PCAP files for this replay be deleted?   [ N ]
   Do you really want to delete this replay?  [ Y ]
   ```

2. Choose whether to delete PCAP files for this replay by typing:
   - `Y` – deletes PCAP files for this replay.
   - `N` – keeps PCAP files. You can manually delete them later.
3. At the confirmation prompt, type `Y` to delete the replay.

8.1.4.4 Analyzing a Replay

Analyze a replay to get a basic replay summary, such as replay duration, number of sessions, number of requests, and number of errors.

Prerequisites

Make sure the PCAP files are accessible. If they are on a remote server, copy them to a local directory that is accessible. Otherwise, `wlacliutil` prompts you to transfer the PCAP files before you can start analyzing a replay.

Context

This version of `wlacliutil` doesn't provide advanced analytical information, such as the longest running requests and the most frequent running requests, or requests from IP, login, or application that sent the most number of requests. Use SAP ASE Cockpit to view the advanced information.

Procedure

1. To begin the analysis, run:

   ```
   analyze start <replay_name>
   ```

   If you don't specify the replay name, you are prompted to enter one.

   The workload analysis begins.
2. `wlacliutil` periodically reports the analysis status. You can also run `analyze status` at any time to check the status. The status report is similar to this:

   ```
   Workload [cmdline_replay5] is in [Identifying requests]. Analysis Percentage:
   [57.17%]. Elapsed Time: [00:00:33]
   ```
3. Run `analyze stop` to stop the workload analysis if necessary.

Results

When the analysis completes, `wlacliutil` displays a report similar to the following:

```
Name                    : cmdline_replay5
Status                  : Analyzed
Start Time              : 2019-01-23 14:33:29
Duration                : 00:08:26
Average Execution Time : 00:00:00.012370
Comments                : This is a demo.
```

```
Number of Sessions     : 6
Number of Requests     : 230
Number of Errors       : 0
Server Name            : 11.163.1.235:13655
```

8.1.5 Comparing Source and Replay Workloads

Compare the workload you captured with a replay workload to identify differences.

Procedure

1. To compare workloads, run:

   ```
   compare start <replay name>
   ```

   If you don't specify the replay name, you are prompted to enter one.
2. `wlacliutil` starts the compare and periodically displays the comparison status.

   ```
   Compare Percentage: [70.00%].
   ```

   You can run `compare status` at any time to check the status.
3. (Optional) Run `compare stop` to stop the comparison.
4. (Optional) Run `compare show <replay_name>` to view the comparison result.

Results

Once the comparison completes, `wlacliutil` displays the comparison result similar to this:

```
                              Capture                                Replay
-------------------------------------------------------------  ----------------
Capture name                  cmdline_capture25                cmdline_replay8
Capture average execution time 00:00:00.012370                 00:00:00.015213
Replay total sessions         6                                6
Replay total requests         230                              230
Capture total errors          0                                0
```

8.1.6 Interactive wlacliutil Commands Reference

You can use the following commands to capture, analyze, and replay workloads in interactive mode.

Some interactive commands are active before you connect to a repository server, while others are active only after you connect to a repository server.

Commands active before connecting to a repository server:

- `connect`

- `exit`
- `help`
- `history`
- `quit`
- `version`
- `!`
- `!!`

Commands active after connecting to a repository server:

- `analyze`
- `capture`
- `compare`
- `disconnect`
- `exit`
- `help`
- `history`
- `quit`
- `replay`
- `version`
- `!`
- `!!`

8.1.6.1 analyze

Analyzes a capture or replay.

Syntax

```
analyze {start | status | stop} [<workload name>]
```

Parameters

`<workload name>`

Name of a captured workload or a replay.

start

Starts to analyze the specified workload.

status

Displays the status of a running analysis.

stop

Stops a workload analysis.

Examples

Example 1

Starts to analyze the workload capture named `server1_cap1`:

```
analyze start server1_cap1
```

Example 2

Displays the status of the running analysis on `server1_cap1`:

```
analyze status server1_cap1
```

Example 3

Ends the running analysis on `server1_cap1`:

```
analyze stop server1_cap1
```

Usage

A repository server must be connected before you run the `analyze` command.

8.1.6.2 capture

Manages a capture on a local or remote SAP ASE server.

Syntax

```
capture {start |stop | status | import | show | delete} [<capture name>]
```

Parameters

<capture name>

Name of the workload capture.

start
: Starts the workload capture.

stop
: Stops the workload capture.

status
: Returns the status of the workload capture.

import
: Imports previously created captured workloads into a repository database.

show
: Displays information for the specified capture. If you don't specify the name, all available captures are listed.

delete
: Removes a captured workload from a repository database.

Examples

Example 1

Starts a workload capture and called it `server1_cap1`:

```
capture start server1_cap1
```

Example 2

Displays the status of the running capture, `server1_cap1`:

```
capture status server1_cap1
```

Example 3

Ends the running capture, `server1_cap1`:

```
capture stop server1_cap1
```

Example 4

Displays all available workload captures:

```
capture show
```

Example 5

Deletes the `server1_cap1` capture:

```
capture delete server1_cap1
```

Example 6

Imports a workload capture named `server1_cap2`:

```
capture import server1_cap2
```

Usage

A repository server must be connected before you run the `capture` command.

8.1.6.3 compare

Compares the workload you captured with a replay workload to identify differences.

Syntax

```
compare {start | status | stop | show} [<replay name>]
```

Parameters

<replay name>
Name of the replay.

start
Starts to compare a replay with its original workload.

status
Displays the status of a running comparison.

stop
Stops a running comparison.

show
Displays the information of a compared replay.

Examples

Example 1

Starts to compare the replay named `cap1_rep1` with its original workload:

```
compare start cap1_rep1
```

Example 2

Displays the status of the running comparison of the `cap1_rep1` replay:

```
compare status cap1_rep1
```

Example 3

Ends the running comparison of the `cap1_rep1` replay:

```
compare stop cap1_rep1
```

Example 4

Displays the comparison results of the `cap1_rep1` replay:

```
compare show cap1_rep1
```

Usage

A repository server must be connected before you run the `compare` command.

8.1.6.4 connect

Connects to an existing repository server.

Syntax

```
connect
```

Usage

A repository server must be connected before you issue other commands.

8.1.6.5 disconnect

Closes the connection to a repository server. The command returns `wlacliutil` to an unconnected state.

Syntax

```
disconnect
```

Usage

Use `connect` to reconnect to the repository server.

8.1.6.6 exit

Exits the `wlacliutil` utility.

Syntax

```
exit
```

Usage

`quit` and Ctrl-C also exit the `wlacliutil` utility.

8.1.6.7 help

Displays information on one or all currently available `wlacliutil` interactive commands.

Syntax

```
help [<command_name>]
```

Parameters

<command_name>

The available interactive command in `wlacliutil`. If you omit `<command_name>`, `help` returns information on all currently available commands.

Examples

Example 1

This example returns information on the `capture` command:

```
help capture
```

Usage

The list of currently available interactive commands changes depending on whether or not `wlacliutil` is connected to a repository server.

8.1.6.8 history

Displays the most recent commands.

Syntax

```
history [<number>]
```

Parameters

<number>

The number of commands to display. If you don't specify a number, `wlacliutil` displays the last 100 history commands.

8.1.6.9 quit

Exits the `wlacliutil` utility.

Syntax

```
quit
```

Usage

`exit` and Ctrl-C also exit the `wlacliutil` utility.

8.1.6.10 replay

Runs replays of captured workloads to measure and analyze application performance.

Syntax

```
replay {start status | stop | show | delete} [<replay name>]
```

Parameters

<replay name>

Name of the replay.

start

Starts a replay of a previously created capture workload.

status
: Displays the status of a running replay.

stop
: Stops a running replay.

show
: Displays information of a replay. If you don't specify a name, `wlacliutil` displays information of all replays.

delete
: Removes a replay from a repository database.

Examples

Example 1

Starts a workload replay named `cap1_rep1`:

```
replay start cap1_rep1
```

Example 2

Displays the status of the running workload replay named `cap1_rep1`:

```
replay status cap1_rep1
```

Example 3

Ends the running replay named `cap1_rep1`:

```
replay stop cap1_rep1
```

Example 4

Displays the analysis results on the `cap1_rep1` replay:

```
replay show cap1_rep1
```

Example 5

Deletes the `cap1_rep1` replay:

```
replay delete cap1_rep1
```

Usage

A repository server must be connected before you run the `replay` command.

8.1.6.11 version

Displays the version of the `wlacliutil` utility.

Syntax

```
version
```

8.1.6.12 Additional Interactive Commands

Additional commands to use within `wlacliutil`.

Command	Description
`!!`	Reissues the last command.
`! <n>`	Reissues the command by its number, which is obtained from the history command.. For example, `!2` reissues the command that is marked as 2.

8.2 wlacliutil With a Response File

Use `wlacliutil` with a response file to capture, analyze, and replay workloads.

A response file contains configuration information for `wlacliutil`. Instead of entering configuration variables when prompted, you save the information in a response file when running a command and then specify the response file name on the `wlacliutil` command line. For example:

```
wlacliutil -S <repository_server_name> -U<repository_username> -
P<repository_password> -i<response_file>
```

The `-S` and `-U` options are required to connect to the repository database when using `wlacliutil` with a resource file.

`wlacliutil` with a resource file only supports the following features :

- Start a capture
- Show a capture/replay
- Import a capture
- Analyze a capture/replay
- List captures/replays

- Delete a capture/replay
- Start a replay

You can specify multiple tasks in a response file. `wlacliutil` performs them sequentially as they appear in the response file. For example, you want to capture a workload from a server and replay it on another server. Then, compare the replay with the original workload. Finally, delete both the capture and the replay. The response file looks like this:

```
task.name=capture
…
task.name=analyze
…
task.name=replay
…
task.name=analyze
…
task.name=compare
…
task.name=delete
…
task.name=delete
…
```

There are a template file and some sample files under the `$SYBASE/WLA/template` (`%SYBASE%\WLA \template` for Windows) directory. You can either create a response file by editing these files or by recording and saving your responses when running `wlacliutil` in interactive mode.

8.2.1 Creating a Response File Using Interactive Mode

Record your inputs in interactive mode to a response file. The response file is a text file that you can edit to change any responses before using it as an input file in subsequent operations.

Context

When running `wlacliutil` in interactive mode, the `-r` command line argument records your responses to the wizard's prompts and creates a response file when the wizard exits.

Procedure

To create the response file when you start `wlacliutil` in interactive mode, run:

```
wlacliutil -S<server_name> -U<repository_username> -P<repository_password> -r
<response_file_name>
```

When specifying the response file name, include its full path. For example:

```
C:\SYBASE\WLA\ResponseFile.txt
```

8.2.2 Creating a Response File from a Sample Response File

You can create a response file by editing a sample response file.

Context

SAP supplies several sample response files for your reference. Each sample contains the attributes that are valid for a certain Workload Analyzer operation. Copy and edit a sample file, which is located under the `$SYBASE/WLA/template/sample` (`%SYBASE%\WLA\template\sample` for Windows) directory.

See wlacliutil in Interactive Mode [page 15] for information about each option in a sample response file.

Procedure

1. Select the sample response file to use.
2. Make a copy of the sample and rename it to distinguish it from the original. For example:

   ```
   cp capture.sample capture.123
   ```

3. Use a text editor to edit the new response file. For example:

   ```
   vi capture.123
   ```

4. Once you have finished editing the response file, start `wlacliutil` from the operating system prompt, using the `-i` flag to specify your response file. For example:

   ```
   wlacliutil -S <repository_server_name>-U<repository_username> -
   P<repository_password> -i<response_file>
   ```

8.2.3 Workload Analyzer Response File Templates

The response file templates for Workload Analyzer are located in the `$SYBASE/WLA/template` (`%SYBASE%\WLA\template` subdirectory).

Template Name	Description
`input.template`	A summary of various Workload Analyzer task templates that you can perform with a response file.
`Capture.sample`	Creates a workload capture.
`Replay.sample`	Replays a captured workload.

Template Name	Description
`Import.sample`	Imports a previously captured workload.
`Analyze.sample`	Analyzes a captured workload or replay.
`Compare.sample`	Compares a source workload with a replay workload.

The following is the `Capture.sample`. The values in your response file may differ from those in this sample response file.

```
################################################################################
###
# This file is used to start a capture with the Workload Analyzer Command
Line     #
#
Utility.
#
################################################################################
###
capture.server.host=10.173.0.245
task.name=capture
################################################################################
###
# Capture Server
information                                                          #
# Note: Value of password must be the encrypted string. A literal password can
be  #
#       encrypted with below
approach:                                                 #
#           WLACLIUtil -e <literal
password>                                          #
#       The output will be the encrypted string. It can be copied as the value
of  #
#
password.                                                                    #
################################################################################
###
capture.server.host=10.173.0.245
capture.server.port=22999
capture.server.user=sa
capture.server.password=2-
AAAAAEbEypqJLmuu2GbrVxEDF6dsQg12dCRJ5SF9rFAiKR0c5fb9s7D7ZhctK8xfS4Y+iw==
capture.server.usessl=default
capture.server.ssl.truststore=c:\work\wkload_cli\cacerts
################################################################################
###
# Capture
options                                                                  #
################################################################################
###
capture.name=cmdline_capture18
capture.comments=This is the first capture from the command line utility.
# Duration must be set in the batch mode. Otherwise, errors will be reported. The
# unit is seconds.
capture.duration=60
# Capture filters setting. The format must be:
#    [include | exclude]: login1, login2, ...,  loginN
#    or
#    [include | exclude]: application1, application2, ...,  applicationN
# The value "default" means all logins|applications will be incldued.
capture.option.filter.login=default
capture.option.filter.application=default
```

```
# The pcap files location. Default value will be under the $SYBASE directory.
capture.option.location=default
# Capture limits. Default value of limit is unlimited (0).
capture.option.limit.filesize=default
capture.option.limit.overhead=default
# Specify whether the full response will be captured. Default value is "No".
capture.option.response.full=default
```

8.3 wlacliutil Command Line Options

Provides a command line interface to use workload analyzer features.

The `wlacliutil` utility is located in:

- (UNIX) `$SYBASE/WLA/bin/wlacliutil.sh`
- (Windows) `%SYBASE%\WLA\bin\wlacliutil.bat`

Syntax

```
wlacliutil [-e <password>]
           [-h]
           [-i <response_file>]
           [-I <interfaces_file>]
           [-L <logfile_name>]
           [-m <message_level>]
           [-P <repository_password>]
           [-r <response_file>]
           [-S <server_name>]
           [-U <repository_username>]
           [-v]
           [-x <truststore_file>]
```

Parameters

-e <password>

generates the encrypted password for the given password. You must use an encrypted password when using `wlacliutil` with a response file.

-h

displays help information for the `wlacliutil` commands that includes options and command syntax.

-i <response_file>

specifies the name of the response file to use for input to `wlacliutil`.

-I <interfaces_file>

specifies the name and location of the interfaces file. If you do not specify `-I`, `wlacliutil` looks for a file named `interfaces` in the directory specified by your SYBASE environment variable.

-L `<logfile_name>`

changes the log file name and location for `wlacliutil`. The default log file is `$SYBASE/WLACLIUtils.log` (`%SYBASE%\WLACLIUtils.log` for Windows).

-m `<message_level>`

specifies which messages are displayed. Valid message levels are:

- 0 – OFF
- 1 – FATAL
- 2 – ERROR
- 3 – WARN
- 4 – INFO
- 5 – DEBUG

`wlacliutil` displays all messages of the level you choose and all messages of greater severity (with lower numbers). That is, if you select message level 3, `wlacliutil` displays messages of level 3, 2, and 1. The default level is 2.

-P `<repository_password>`

is the password to connect to the repository server. If you do not specify the `-P` flag, `wlacliutil` prompts for a password. If you use the `-P` flag without an argument, `wlacliutil` assumes a NULL password.

-r `<response_file>`

saves the user inputs into the specified file that can be used as the response file.

-S `<server_name>`

specifies the repository server name. You can use the corresponding server entry in the interfaces file or specify the server name in the `<host:port>` format.

-U `<repository_username>`

specifies the login name to connect to the repository server. Login names are case sensitive.

-v

displays the `wlacliutil` version and copyright.

-x `<truststore_file>`

specifies the secure sockets layer (SSL) TrustStore file that is used to establish SSL connection to the repository SAP ASE server. SSL is enabled if this option is specified.

Examples

Example 1

Starts `wlacliutil` in interactive mode:

```
$SYBASE/WLA/Bin/wlacliutil -S 10.173.0.245:22999 -U sa -P Password123
```

Example 2

Starts `wlacliutil` with a response file:

```
$SYBASE/WLA/Bin/wlacliutil -S 10.173.0.245:22999 -U sa -P Password123 -i
capture.sample
```

Example 3

Encrypts the given password for a repository server:

```
$SYBASE/WLA/Bin/wlacliutil  -e Password123
```

Usage

- When `-i` is specified, `wlacliutil` works with a response file. It reads the provided response file and performs the specified actions automatically. In this case, the `-S` and `-U` options are mandatory. The `-P` option is optional. If it is not specified, `wlacliutil` prompts you to enter a valid password for the repository database after `wlacliutil` starts to run.
- When `-i` is not specified, `wlacliutil` works in interactive mode. It reads the your input commands and returns the results to the command line. In this case, the `-S` and `-U` options are mandatory. The `-P` and `-I` options are optional. If the `-P` option is not specified, `wlacliutil` prompts you to enter a valid password for the repository database after `wlacliutil` starts to run.
- When the `-I` option is specified, `wlacliutil` searches the server entry in the specified interfaces file. If it is not specified, the utility checks the LDAP settings. If LDAP is not specified or the entry is not found in the LDAP sever, `wlacliutil` searches the `$SYBASE/interfaces` file by default. If the server entry is still not found, then `wlacliutil` fails to start.
- When `-x` is specified, `wlacliutil` connects to the repository server with SSL enabled. The TrustStore file includes the public key used to establish SSL enabled connections. Import the public key into this TrustStore file with the following command:

  ```
  $SAP_JRE/bin/keytool.exe -import -keystore <TrustStore> -storepass changeit -
  file trusted.txt
  ```

9 Capturing, Analyzing, and Replaying Workloads with SAP ASE Cockpit

Workload capture and replay allows you to capture workloads and monitor SQL queries sent to SAP ASE to analyze and measure database server activity.

Activity	Description
Capture	Enable the `enable workload analyzer` configuration parameter on the SAP ASE server in order to capture workload on that server. There are two ways you can capture a workload: • The `dbcc workload_capture` in a SAP ASE server • The Workload Capture wizard in the SAP ASE cockpit **i Note** Before starting a capture, dump the database related to the capture and save the server configurations files to use to set up the replay test environment.
Analyze	Once you capture a workload, the dashboard in the SAP ASE cockpit provides graphical reports for your capture. From the dashboard, you can view summary information such as: • Capture duration. • Number of sessions. • Number of requests. • Number of errors. • Top long-running requests. • Top frequent-running requests. • Reports that identify frequent requests base on IP address, login, and application.
Replay	After you capture a workload from an SAP ASE production server, you can replay it via the SAP ASE cockpit, in a test environment with similar attributes as its original workload. This allows you to analyze the results and performance impact on the test system without actually changing the SAP ASE production server. Before starting a replay, manually set up a test environment: • Load the database to the test server that is dumped before capture. • Restore configurations for the test server. • Create required logins and privileges. Do not use a comma (`,`) as part of the replay server's password. If you do, the replay sessions fail to connect to the replay server. • Create the required database user on the test server.

Activity	Description
	• Configure the charset for test server so that it is the same as that used for workload capture. • Configure the number of user connections for the test server to be the same as that for workload capture. **i Note** The workload that is captured during replay will include all activity that takes place on the replay server during the replay session. If other applications besides SAP ASE Cockpit are connected to the server, then the activity of these applications are included. This activity may also affect the behavior of the replayed workload. To avoid this, ensure that no other applications besides SAP ASE Cockpit are connected to the replay server during the replay session.

See the *SAP Adaptive Server Enterprise Cockpit* documentation for complete information about using this option.

9.1 Set the Default Network Packet Size

Set the value of the default network packet size on the replay server.

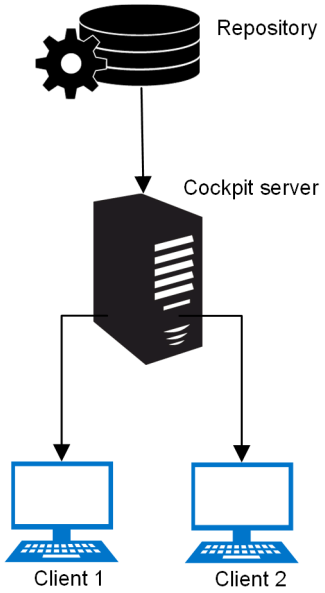
Set the value of the `default network package size` configuration parameter on the replay server so that it is greater than 608 and not smaller than that used on the capture server. This requirement does not affect the repository server or the production (capture) server.

In some cases the workload analyzer creates login packets for connections involved in the replay process. The size of the login packet created is 608 bytes. This is only used on the replay server.

9.2 Restriction Considerations

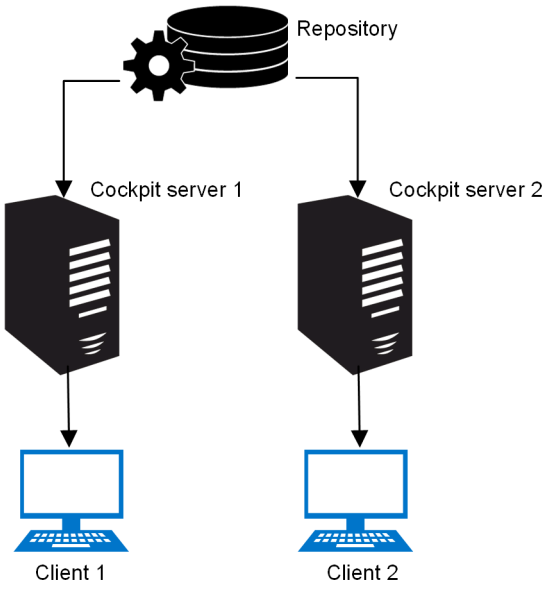
Consider how you plan to analyze your workload, as there are some restrictions to what activities are available depending on your setup.

The following scenario uses a repository database, one cockpit server, and two clients sharing the single cockpit server.



If Client #1 is:	Can Client #2 Capture?	Can Client #2 Analyze?	Can Client #2 Replay?
Capturing	No	Yes	No
Analyzing	Yes	No	No
Replaying	No	No	No

This second scenario uses a repository database, two cockpit servers, and two clients that each has its own separate connection to a cockpit server.



If Client #1 is:	Can Client #2 Capture?	Can Client #2 Analyze?	Can Client #2 Replay?
Capturing	Yes	Yes	Yes

If Client #1 is:	Can Client #2 Capture?	Can Client #2 Analyze?	Can Client #2 Replay?
Analyzing	Yes	No	No
Replaying	Yes	No	No

9.3 dbcc workload_capture

The workload_capture parameter for the database consistency checker (dbcc) provides the ability to enable and disable workload captures, and configure workload capture settings.

Syntax

```
dbcc workload_capture (start)
```

```
dbcc workload_capture (stop)
```

```
dbcc workload_capture (mode[, <mode_value>])
```

```
dbcc workload_capture (dir[, <dir_name>])
```

```
dbcc workload_capture (limit[, overhead | space | time[, <limit_value>]])
```

```
dbcc workload_capture (preference[, (logins | nologins) | (appls | noappls)
    | full_response | stoperrors | skip_thisconn[, <pref_value>]])
```

```
dbcc workload_capture (reset, [dir] | [mode] | [limit[, overhead | space |
time]] |
    [preference[, [logins | nologins] | [appls | noappls] |
    full_response | stoperrors | merge | compress | hide_password |
skip_thisconn])
```

```
dbcc workload_capture (status)
```

```
dbcc workload_capture (help)
```

Parameters

start

Starts the workload capture.

> **Note**
>
> The commands submitted by the connection that executes `dbcc workload_capture (start)` are not captured unless you change the default value of the `skip_thisconn` option to 0.

stop

Stops the workload capture.

mode[, <mode_value>]

Sets different capture mode. The valid values of `<mode_value>` are:

- `normal` – (default) normal mode.
- `simulate` – simulation mode, in which capturing is simulated by doing everything except writing to files. Output size can thus be determined, as well as an approximation of overhead.

dir[, <dir_name>]

Specifies a directory to which the capture is saved. This must be an absolute path.

> **Note**
>
> When you specify a directory name, the pcap files for all subsequent captures are saved to this directory. This means that pcap files from multiple captures can be generated and stored in the same directory. The file names contain the server name, engine number and timestamp indicating when the capture was performed. The engine number is significant because there will be one PCAP file per engine thread.

limit[, overhead | space | time[, <limit_value>]]

Sets different limits, at which point the SAP ASE server stops capturing workloads:

Options	Description
`overhead`	Specifies the acceptable overhead percentage. • The valid values for `<limit_value>` is 0–100 • The default value is 10, which means that the workload capture time cannot exceed 10 percent of the total execution time.
`space`	Specifies the maximum capture file size in megabytes. If multiple files are generated for the capture, the total size of all captured files should not exceed this limit. The default value of `<limit_value>` is 1024 MB. **Note** Files can grow very large, very quickly. The existence of a size limit helps prevent a file system from becoming full. If larger pcap files are required, increase the value of the space limit.

Options	Description
`time`	Specifies the capture time limit in seconds. The capture stops when the capture time exceeds the limit you specify in `<limit_value>`. The default value of `<limit_value>` is 0, which means there is no time limit.

Specifying the `<limit_value>` as `0` turns off the limit for that specific limit.

> **Note**
>
> When a capture is running, `limit` is the only settings you can change. If the current capture reaches the new limit, the capture stops immediately.

preference[, (logins | nologins) | (appls | noappls) | full_response | stoperrors | merge | compress | skip_thisconn[, <pref_value>]]

Set these preferences for the capture:

Options	Description

`logins` | `nologins`[, `<pref_value>`]	Specify:
	
	• `logins` – to configure SAP ASE to capture only the workloads of the logins you specify in `<pref_value>`.
	• `nologins` – to exclude the workloads from the logins you specify in `<pref_value>` from capture.
	
	Separate multiple logins with commas in `<pref_value>` using the string format `"user1, user2, user3"`. Explicitly specifying `"all"` is the same as not specifying this option, in which the workload for all logins are captured.
	
	If you do not specify `<perf_value>`, ASE returns the current user settings, such as the following examples:
	
	`Workload capture enabled logins: user1, user2, user3`
	
	`Workload capture enabled logins: all`

`appls` | `noappls`[, `<pref_value>`]	Specify: • `appls` – to configure SAP ASE to capture only the workload of the applications you specify in `<pref_value>`. • `noappls` – to exclude the workloads from the applications you specify in `<pref_value>` from capture. Separate multiple applications with commas in `<pref_value>` using the string format `"appl1, appl2, appl3"`. Explicitly specifying `"all"` is the same as not specifying this option, in which the workload for all applications are captured.

Options	Description

	If you do not specify <perf_value>, ASE returns the current user settings, such as the following examples:
	`Workload capture enabled applications: appl1, appl2`
	`Workload capture enabled applications: all`

full_response[, <pref_value>]	Allows you to specify whether to capture all TDS data. The valid values for <perf_value> are:
	• 1 – (Default) All TDS data is captured.
	• 0 – Only TDS_DONE packets are captured for response TDS.
	If you do not specify <perf_value>, ASE returns the current settings, such as the following examples::
	`Workload capture full response: yes`
	`Workload capture full response: no`

stoperrors[, <pref_value>]	Instructs ASE to stop capturing a workload when it encounters the server errors you specify in <pref_value>.
	Separate multiple error numbers with commas in <pref_value> using the string format `"error1, error2, error3"`.
	If you do not specify <perf_value>, ASE returns the current user settings, such as the following examples:
	`Workload capture stoperrors: 208, 1024`
	`Workload capture stoperrors: none`
	i Note
	Do not specify errors other than SAP ASE server errors, such as kernel errors.

skip_thisconn[, <pref_value>]	Allows you to specify whether to omit the requests submitted by the current connection when starting or stopping a workload capture task. The valid values for <perf_value> are:
	• 1 – (Default) skips the current connection that issues the `dbcc workload_capture` command.
	• 0 – does not skip the connection.
	If you do not specify <perf_value>, ASE returns the current settings, such as the following examples:
	`Workload capture skip this connection: yes`
	`Workload capture skip this connection: no`

Options	Description
	Note The commands submitted by the connection that executes `dbcc workload_capture (start)` are not captured unless you change the specify the value of skip_thisconn value as 0.

Executing `dbcc workload_capture (preference)` without specifying any options returns all preference settings:

reset, [dir] | [mode] | [limit[, overhead | space | time]] | [preference[, [logins | nologins] | [appls | noappls] | full_response | stoperrors | merge | compress | hide_password | skip_thisconn]

Resets any of the parameters you configured, to the default setting.

status

Returns the status of the workload capture, in the following format:

```
Workload capture status: <state>
Stop reason: <motive>
Elapsed time: <n> sec
File size: <n> bytes
Overhead: <n> sec, <n>%
Capture file: <file1>
Capture file: <file2>
```

where:

- `<state>` – is `idle`, `started`, `starting`, `stopped`, or `stopping`.
- `<motive>` – is `reboot`, `request`, `time limit`, `size limit`, `overhead limit`, `error`, `rapidlog`, `unavailable`, `n/a`

help

Displays the usage of this command.

Examples

Setting all configurations

This example is a typical use scenario that sets all of the settings at once:

```
> dbcc workload_capture (limit, overhead, 15)
> go
> dbcc workload_capture (limit, space, 4096)
> go
> dbcc workload_capture (limit, time, 0)
> go
> dbcc workload_capture (dir, '/tmp')
> go
> dbcc workload_capture (preference, logins, 'user1, user2, user3')
> go
> dbcc workload_capture (preference, appls, 'DBACockpit, ISQL')
> go
> dbcc workload_capture (start)
```

```
> go
...
> dbcc workload_capture (stop)
> go
```

Obtaining Status Information

This example returns the status information for a completed capture:

```
> dbcc workload_capture (status)
> go
```

```
Workload capture status: disabled
Stop reason: request
Elapsed time: 413 sec
File size: 40 bytes
Overhead: 10 sec, 8%
Capture file: /tmp/mycapture.pcap
```

This example returns the status information for an ongoing capture:

```
Workload capture status: started
Elapsed time: 300 sec
File size: 423 MB
Overhead: 20 Seconds, 8%
Capture file: /opt/sybase/mypcap.pcap.0001
Capture file: /opt/sybase/mypcap.pcap.0002
```

Storage Location

This example specifies where to save the PCAP file to:

```
dbcc workload_capture (dir, '/workload/capture')
```

Setting Limits

This example sets the overhead limit to 20 percent:

```
dbcc workload_capture (limit, overhead, 20)
```

This example sets the space limit to 4096 MB:

```
dbcc workload_capture (limit, space, 4096)
```

This example sets the time limit to 3600 seconds:

```
dbcc workload_capture (limit, time, 3600)
```

This example executes workload capture limits without specifying the different limit types or values, so that all limit settings are returned:

```
> dbcc workload_capture (limit)
> go
```

```
Workload capture overhead limit: 10%
Workload capture space limit: 1024MB
Workload capture time limit: 600s
```

Filtering Logins

This example applies a login filter to capture activity only for logins user1 and user2:

```
dbcc workload_capture (preference, logins, 'user1, user2')
```

This example applies a filter to exclude the capture of activities by the sa login:

```
dbcc workload_capture (preference, nologins, 'sa')
```

Filtering Applications

This example applies an application filter to capture activity only from `isql` and `bcp`:

```
dbcc workload_capture (preference, appls, 'isql, bcp')
```

This example applies a filter to exclude the capture of activities from SCC and the ASE Agent plugin::

```
dbcc workload_capture (preference, noappls, 'SCC_ASE, ASEAgentPlugin')
```

Standards

ANSI SQL – Compliance level: Transact-SQL extension.

Permissions

The SAP ASE server must have operating system permission to open any capture files.

The permission checks for `dbcc workload_capture` differ based on your granular permissions settings.

Setting	Description
Enabled	Only users with `set tracing any process` permission can active capturing.
Disabled	Only users with SA or SSO role can activate capturing.

10 Capturing, Analyzing, and Replaying Workloads in SAP ASE 15.7

You can use SAP ASE 15.7 SP137 to capture a workload executed on your SAP ASE server and then analyze and replay the workload using the 16.0 SP02 SAP ASE cockpit. This can help you discover any changes to your database design or server configuration that can improve performance in 16.0 SP02.

> **i Note**
>
> All references to 15.7 SP137 indicate version 15.7 SP137 and later, and all references to 16.0 SP02 indicate version 16.0 SP02 and later.

If you are considering upgrading your applications to 16.0 SP02, you can capture your workload on a 15.7 SP137 server and replay your application workload against a 16.0 SP02 server. You can then and compare the performance of your application on the 15.7 SP137 and 16.0 SP02 servers.

Additionally, if you have a 16.0 SP02 SAP ASE cockpit and are not considering upgrading your application to that version, you can still capture your workload in your 15.7 SP137 server and analyze it using the 16.0 SP02 SAP ASE cockpit.

The workload capture functionality is enabled using the `enable workload analyzer` configuration parameter. You can start and stop the capture process using the `dbcc workload_capture` command. Workloads are captured as PCAP files which can be imported into the 16.0 SP02 SAP ASE cockpit so that you can view and analyze your workload in detail using the cockpit dashboards and wizards.

> **i Note**
>
> You do not need an `ASE_WORKLOADANALYZER` license to use the workload analyzer on version 15.7 SP137.

Activity	Description

Capture	1. On the 15.7 SP137 server for which you want to capture workloads, set the `enable workload analyzer` configuration parameter to `1` (on).
	2. Use these `dbcc workload_capture` commands to start and stop the capture to create PCAP files of your workload:
	`dbcc workload_capture(start)`
	`dbcc workload_capture(stop)`
	i Note
	See dbcc workload_capture [page 48] for complete syntax, including information about PCAP location configuration, and permissions. Permissions differ depending on whether granular permissions is enabled on your server.

Activity	Description
	See Capturing, Analyzing, and Replaying Workloads with SAP ASE Cockpit [page 45] and *Capture an SAP ASE Workload* in the *SAP Adaptive Server Enterprise Cockpit* guide for more information.
Analyze	Import your PCAP files into the 16.0 SP02 SAP ASE cockpit in using the Workload Analyzer Import Wizard. The dashboard in the SAP ASE cockpit provides graphical reports for your capture and summary information such as: • Capture duration • Number of session. • Number of requests • Number of errors • Top long-running requests • Top frequent-running requests • Reports that identify frequent requests based on IP address, login, and application See *Importing a Captured Workload into a Repository Database*and *Analyzing Captured Workloads* in the *SAP Adaptive Server Enterprise Cockpit* guide for more information.
Replay	After you capture a workload from the 15.7 SP137 production server, you can replay it in a test environment with similar attributes to those of the server used for the original workload using the 16.0 SP02 SAP ASE cockpit. This allows you to analyze the results and performance impact on the test system without actually changing the production server. See Capturing, Analyzing, and Replaying Workloads with SAP ASE Cockpit [page 45] and *Replay a Captured Workload* in the *SAP Adaptive Server Enterprise Cockpit* guide for more information.

Related Information

SAP Adaptive Server Enterprise Cockpit

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering a SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Gender-Related Language

We try not to use gender-specific word forms and formulations. As appropriate for context and readability, SAP may use masculine word forms to refer to all genders.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

www.sap.com/contactsap

THE BEST RUN SAP